

Timeline

Timeline The Timeline component is a powerful and flexible way to display chronological events, project milestones, or any sequential information. It provides multiple orientations, status indicators, and extensive customization options while maintaining accessibility and responsive design. The <aava-timeline> component supports both vertical and horizontal layouts, various text alignment patterns, multiple sizes, and status-based visual indicators. It's perfect for project management, user onboarding flows, historical displays, and any scenario requiring chronological representation.

How to use ■ import { AavaTimeline , TimelineEvent } from "@aava/play-core" ;

Basic Timeline ■ A simple vertical timeline with default settings.

Text Alignment Variants ■ Different text alignment patterns for visual variety.

Orientations ■ Vertical and horizontal timeline layouts.

Size Variants ■ Multiple size options for different use cases.

API Reference ■

Input Properties	Property Type	Default	Description
events	TimelineEvent[]	[]	Array of timeline events to display
orientation	'vertical' 'horizontal'	'vertical'	Timeline layout direction
size	'x-small' 'small' 'medium' 'large'	'large'	Overall size of the timeline
sortOrder	'ascending' 'descending'	'ascending'	Order to sort events
textAlign	'zig-zag' 'zigzag-left' 'zigzag-right' 'zigzag-up' 'zigzag-down' 'zigzag-left'	'zigzag-left'	Text alignment pattern
iconCircleBgColor	string	'#fff'	Background color for icon circles
iconCircleBorderColor	string	'#ccc'	Border color for icon circles
iconCircleSize	string	'40px'	Size of icon circles
disabled	boolean	false	Disables the entire timeline
customSort	(a: TimelineEvent, b: TimelineEvent) => number	undefined	Custom sorting function
contentTemplate	any	undefined	Template for custom content projection

TimelineEvent Interface ■

```
interface TimelineEvent {
  time ? : string ; // Time of the event
  text ? : string ; // Short text description
  iconName ? : string ; // Icon name from icon library
  iconColor ? : string ; // Custom icon color
  iconSize ? : string ; // Custom icon size
  imageUrl ? : string ; // Image URL for custom images
  imageSize ? : string ; // Custom image size
  year ? : string ; // Year of the event
  title ? : string ; // Event title
  description ? : string ; // Detailed description
  status ? : "completed" | "current" | "pending" | "unreached" ;
  customClass ? : string ; // Additional CSS classes
  data ? : any ; // Custom data for advanced use cases
}
```

Features ■

Multiple Orientations ■

- Vertical : Traditional top-to-bottom timeline layout
- Horizontal : Left-to-right layout for wide screens

Text Alignment Patterns ■

- zig-zag : Alternating left/right alignment for vertical, up/down for horizontal
- zigzag-left : All text aligned to the left
- zigzag-right : All text aligned to the right
- zigzag-up : All text aligned upward (horizontal only)
- zigzag-down : All text aligned downward (horizontal only)

Status System ■

- completed : Finished events with success styling
- current : Currently active events with highlight styling
- pending : Upcoming events with neutral styling
- unreached : Future events with disabled styling

Size Variants ■

- x-small : Compact size for dense interfaces
- small : Small size for limited space
- medium : Standard size for most use cases
- large : Large size for prominent displays

Customization Options ■

- Custom icon and image support
- Flexible color theming
- Custom sorting functions
- Content projection for advanced layouts
- Responsive design with automatic spacing

Best Practices ■

- Content Organization ■ Use clear, descriptive titles for each event
- Keep descriptions concise but

informative Use consistent date/time formats Group related events logically Visual Hierarchy ■ Choose appropriate sizes based on content importance Use status indicators consistently across events Maintain visual balance with text alignment patterns Consider screen size when choosing orientation Accessibility ■ Provide meaningful alt text for images Use semantic status indicators Ensure sufficient color contrast Support keyboard navigation Performance ■ Limit the number of events for optimal performance Use custom sorting functions sparingly Consider lazy loading for large timelines Optimize images and icons Project Timeline ■

```
const projectTimeline : TimelineEvent [ ] = [ { year : "2023 Q1" , title : "Planning Phase" , description : "Requirements gathering and project planning" , status : "completed" , iconName : "check-circle" , } , { year : "2023 Q2" , title : "Development" , description : "Core feature development" , status : "current" , iconName : "code" , } , { year : "2023 Q3" , title : "Testing" , description : "Quality assurance and testing" , status : "pending" , iconName : "bug" , } , { year : "2023 Q4" , title : "Launch" , description : "Production deployment" , status : "unreached" , iconName : "rocket" , } , ] ;
```

User Journey Timeline ■

```
const userJourney : TimelineEvent [ ] = [ { time : "0 min" , title : "Welcome" , description : "User lands on homepage" , status : "completed" , iconName : "home" , } , { time : "2 min" , title : "Browse" , description : "User explores content" , status : "current" , iconName : "search" , } , { time : "5 min" , title : "Engage" , description : "User interacts with features" , status : "pending" , iconName : "heart" , } , ] ;
```

Styling ■ The timeline component uses CSS custom properties for theming and supports extensive customization through: Icon circle colors and sizes Status-based color schemes Responsive spacing adjustments Custom CSS classes for advanced styling

```

<aava-timeline
  [events]="basicEvents"
  [size]="'large'"
  [orientation]="'vertical'"
  [textAlign]="'zigzag-left'"
  [iconCircleSize]="'40px'"
  [iconCircleBgColor]="'#fff'"
  [iconCircleBorderColor]="'#ccc'"
></aava-timeline>

```

```

basicEvents: TimelineEvent[] = [
  {
    time: '10:00 AM',
    title: 'Project Kickoff Meeting',
    description: 'Initial project planning and team alignment session',
    iconName: 'circle-check',
    iconColor: '#E91E63',
    iconSize: '24px',
    year: '2025',
    status: 'completed',
  },
  {
    time: '2:00 PM',
    title: 'Development Phase Started',
    description: 'Begin coding and implementation of core features',
    iconName: 'loader',
    iconColor: '#E91E63',
    iconSize: '20px',
    year: '2025',
    status: 'current',
  },
  {
    time: '4:30 PM',
    title: 'First Milestone Completed',
    description: 'Successfully completed the initial development phase',
    iconName: 'check-circle',
    iconColor: '#E91E63',
    iconSize: '16px',
    year: '2025',
    status: 'pending',
  },
];

selectedEvent: TimelineEvent | null = null;

onEventSelect(event: TimelineEvent): void {
  console.log('Timeline event selected:', event);
  this.selectedEvent = event;
}

```

```

<aava-timeline
  [events]="alignmentEvents"
  [size]="'large'"
  [orientation]="'vertical'"
  [textAlign]="'zigzag-left'"
  [iconCircleSize]="'40px'"
  [iconCircleBgColor]="'#fff'"
  [iconCircleBorderColor]="'#ccc'"
></aava-timeline>

```

```

<aava-timeline
  [events]="alignmentEvents"
  [size]="'large'"
  [orientation]="'vertical'"
  [textAlign]="'zigzag-right'"
  [iconCircleSize]="'40px'"
  [iconCircleBgColor]="'#fff'"
  [iconCircleBorderColor]="'#ccc'"
></aava-timeline>

```

```

<aava-timeline
  [events]="alignmentEvents"
  [size]="'large'"
  [orientation]="'vertical'"
  [textAlign]="'zig-zag'"
  [iconCircleSize]="'40px'"
  [iconCircleBgColor]="'#fff'"
  [iconCircleBorderColor]="'#ccc'"
></aava-timeline>

```

```

alignmentEvents: TimelineEvent[] = [
  {
    time: '9:00 AM',
    title: 'Morning Standup',
    description: 'Daily team synchronization meeting',
    iconName: 'circle-check',
    iconColor: '#E91E63',
    iconSize: '24px',
    year: '2025',
    status: 'completed',
  },
  {
    time: '11:30 AM',
    title: 'Code Review',
    description: 'Peer review of new feature implementation',
    iconName: 'check',
    iconColor: '#E91E63',
    iconSize: '20px',
    year: '2025',
    status: 'current',
  },
  {
    time: '2:00 PM',

```

```

        title: 'Testing Phase',
        description: 'Quality assurance and bug testing',
        iconName: 'circle-check',
        iconColor: '#E91E63',
        iconSize: '16px',
        year: '2025',
        status: 'pending',
    },
    {
        time: '4:30 PM',
        title: 'Deployment',
        description: 'Production deployment and monitoring',
        iconName: 'rocket',
        iconColor: '#E91E63',
        iconSize: '18px',
        year: '2025',
        status: 'unreached',
    },
];

selectedEvent: TimelineEvent | null = null;

onEventSelect(event: TimelineEvent): void {
    console.log('Timeline event selected:', event);
    this.selectedEvent = event;
}

```

■ No code found

■ No code found